Practical – 3

Aim: Implement and analyse algorithms given below:

3.1) Implement and analyze Quick Sort Algorithm

Program Code:

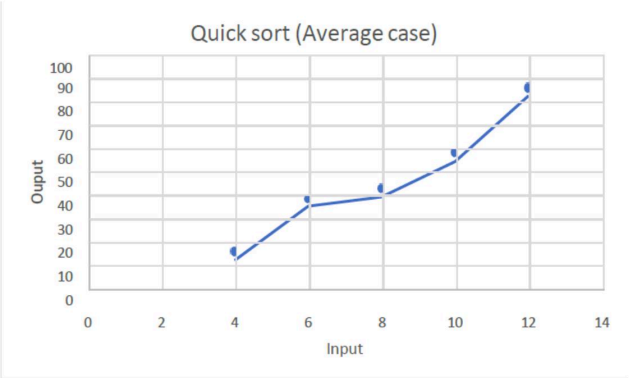
```
import java.util.*;
import java.time.*;
public class trial
{
   public static void QuickSort(int a[],int starting,int ending)
   {
      if(starting>=ending)
      {
         return;
      }
      int pivot_index=partition(a,starting,ending);
      QuickSort(a, starting, pivot_index-1);
      QuickSort(a, pivot_index+1, ending);
   }
   public static int partition(int a[],int starting,int ending)
   {
      int pivot=a[ending];
      int i=starting-1;
      for(int j=starting ; j<=ending ; j++)
      {
         if(a[j]<pivot)
```

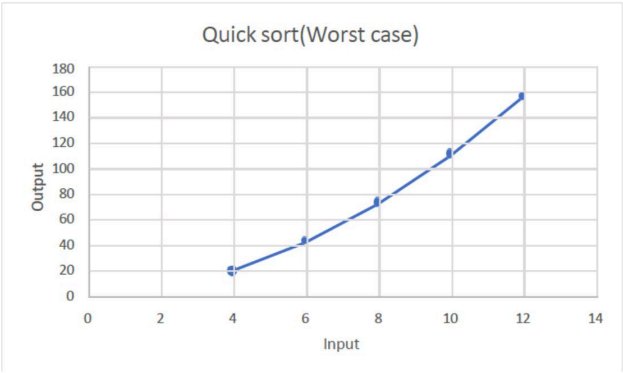
```
            {
                i++;
                int temp=a[j];
                a[j]=a[i];
                a[i]=temp;
            }
        }
        i++;
        int temp=pivot;
        a[ending]=a[i];
        a[i]=temp;
        return i;
    }
    public static void display(int a[])
    {
        System.out.println("Array: ");
        for(int i=0 ; i<a.length ; i++)
        {
            System.out.print(" "+a[i]);
        }
        System.out.println();
    }
public static void main(String[] args)
    {
        System.out.println("\nQuick Sort");
```

```
        int c[]={5,2,6,7,1,3,4};
        Instant inst1 = Instant.now();
        QuickSort(c,0,c.length-1);
        Instant inst2 = Instant.now();
        System.out.println("Elapsed Time: "+ Duration.between(inst1, inst2).toString());
        display(c);}
}
```

Output:

```
Quick Sort
Elapsed Time: PT0.000009024S
Array:
 1 2 3 4 5 6 7
```





Conclusion: From this practical I learned about quick sort technique and how it used divide and conquer strategy for sorting the data. Time complexity is O(nlogn).Quick Sort technique is efficient on large datasets.

Staff Signature:

Grade:

Remarks by the Staff:

Practical – 4

Aim: Implement and analyse algorithms given below:

4.1) A Burglar has just broken into the Fort! He sees himself in a room with n piles of gold dust. Because each pile has a different purity, each pile also has a different value (v[i]) and a different weight (w[i]). A Burglar has a bag that can only hold W kilograms. Calculate which piles Burglar should completely put into his bag and which he should put only fraction into his bag. Design and implement an algorithm to get maximum piles of gold using given bag with W capacity, Burglar is also allowed to take fractional of pile.

Program Code:

Output:

4.2) Implement the program to find the shortest path from one source to all other destinations in any city graph.

Program Code:

Output:

4.3) Find Minimum Cost spanning tree of an undirected graph using Kruskal's algorithm.

Program Code:

Output:

Conclusion:

Staff Signature:

Grade:

Remarks by the Staff: